

Kurs: Programming Tools		Code: W2PROG20
Studiengang	BSc Wirtschaftsinformatik	
Studiengruppe	BWI-A21	
Dozent/in	Walter Rothlin	
Leistungsnachweis(e)	Ein Einzelarbeit mit Fachgespräch	
Präsenz	Selbststudium	Workload
36 h	56 h	92 h

## Lernergebnisse

Die Studierenden

- erlernen vertiefender Konzepte im OO Programmierung mittels Python
- können nach Abschluss des Kurses selbstständig Python-Programme für kleinere und mittlere Probleme nach OO Design-Patterns entwickeln
- haben die Fähigkeit, Python Code zu lesen, zu analysieren und zu verbessern (refactoring)

## Lehrinhalte

1. Klassen, Instance-Variablen und Methoden / Vererbung
2. Sichtschutz / Scope von Variablen und Methoden steuern
3. Functions: Positional- und Named-Parameter mit Default Werten
4. Exception-Handling in Python / Eigene Exceptions
5. Datenstrukturen und Containers
6. GUI-Programmierung
7. GUI-Programmierung (continue)
8. Daten aufbereiten und auswerten (Numpy, Matplotlib, Pandas, Excel)
9. AI in Python nutzen: OCR (optical character recognition)

## Lehr- und Lernmethoden

- Stoffvermittlung Frontalunterricht, Übungen lösen im Präsenzunterricht und Selbststudium

## Leistungsnachweis

Eine Applikation anhand von vorgegebenen Requirements nach den Clean-Code Regeln in Einzelarbeit entwickeln.

Geprüft wird der Code anhand der funktionalen und Qualitäts-Kriterien. Anhand eines Fachgespräches wird überprüft, ob der Student den Code und die darunter liegenden theoretischen Grundlagen verstanden hat. (Gewicht: 100%)

## Lehrmittel

### Verpflichtend

- Python 3 – Einsteigen und Durchstarten: Python lernen für Anfänger und Umsteiger. Inkl. Kapiteln zu Git und Minecraft Pi. Inkl. E-Book

## Semesterprogramm

### 1) Klassen, Instance-Variablen und Methoden, Vererbung

Datum	Di 27.09.2022
Zeit	17:30 – 20:45
Dozent	Walter Rothlin

#### Vorbereitung

- Python Grundlagen repetieren und IDE (PyCharm) Basics beherrschen
- Buch Python 3 (Einsteigen und Durchstarten): Kapitel 5.1 und 5.2 durcharbeiten

#### Inhalte

- Eine eigene (z.B. Bruch, IncDec) Klasse in Python implementieren und Testen
- API-Doc der Klasse mit Doc-String erstellen

### 2) Sichtschutz / Scope von Variablen und Methoden steuern

Datum	Di 11.10.2022
Zeit	17:30 – 20:45
Dozent	Walter Rothlin

#### Vorbereitung

- Buch Python 3 (Einsteigen und Durchstarten): Kapitel 5.3, 5.4, 5.5 und 5.6 durcharbeiten

#### Inhalte

- Zugriff auf Attribute respektive Unterschied zwischen Properties und Attribute
- Dynamische Attribute in Python
- Klassenattribute
- Statische Methoden

#### Nachbereitung

- Refactoring des eigenen Codes und die Verwendung von Properties anstelle von Attributen.
- Testen der Sichtbarkeit

3) Functions: Positional- und Named-Parameter mit Default Werten	
Datum	Do 13.10.2022
Zeit	13:15 – 16:30
Dozent	Walter Rothlin

#### Vorbereitung

- Buch Python 3 (Einsteigen und Durchstarten): Kapitel 4 durcharbeiten

#### Inhalte

- Eine Library mit eigenen Funktionen erstellen (readInt / readFloat) und ein Test-Programm schreiben, um Test-Cases automatisch auszuführen.
- Parameter der Funktionen erweitern, so dass diese abwärtskompatible bleiben (Bestehende Calls funktionieren noch)
- ➔ Named-Parameter beim Aufruf und Default-Werte den Argumenten zuordnen
- DOC-String zu den eigenen Funktionen

#### Nachbereitung

- Analysieren und Verwendung der Format-Funktionen (Methoden) bei der String-Formatierung.

4) Exception-Handling in Python / Eigene Exceptions	
Datum	Di 25.10.2022
Zeit	17:30 – 20:45
Dozent	Walter Rothlin

#### Vorbereitung

- Buch Python 3 (Einsteigen und Durchstarten): Kapitel 7 durcharbeiten

#### Inhalte

- Try-except Konstrukt mit else und finally
- Raise exception
- Exception Hierarchie, Eigene Exceptions

#### Nachbereitung

- Library functions auf Exceptions erweitern.

5) Datenstrukturen und Containers	
Datum	Di 15.11.2022
Zeit	17:30 – 20:45
Dozent	Walter Rothlin

#### Vorbereitung

- Buch Python 3 (Einsteigen und Durchstarten): Kapitel 6 durcharbeiten

#### Inhalte

- Listen, Tuples, Dictionaries und Sets, JSON-Strukturen
- Strings verarbeiten
- Muster Erkennung (RegEx)
- REST-Call und JSON parsen
- **Abgabe der Aufgabenstellung und Kriterien für den Leistungsnachweis. Abgabe des Leistungsnachweises bis 2.12.21 / 18:00.**

#### Nachbereitung

- **Leistungsnachweis erstellen**

6) GUI Programmierung	
Datum	Di 13.12.2022
Zeit	17:30 – 20:45
Dozent	Walter Rothlin

#### Vorbereitung

- Vorgegebenes Beispiel auf eigenem Pycharm zum Laufen bringen (inkl. Installation der nötigen Libraries).

#### Inhalte

- GUI Elemente (Label, Button, Text-Box, Check- und Radio-Buttons, List-Boxen)
- Dialogfenster
- Layout-Managers
- Events / Event-Handling

#### Nachbereitung

- GUI-Applikation (z.B. Wetter App) entwickeln

7) GUI Programmierung (continue)	
Datum	Di 10.01.2023
Zeit	17:30 – 20:45
Dozent	Walter Rothlin

#### Vorbereitung

- Vorgegebenes Beispiel auf eigenem Pycharm zum Laufen bringen (inkl. Installation der nötigen Libraries).

#### Inhalte

- GUI Elemente (Label, Button, Text-Box, Check- und Radio-Buttons, List-Boxen)
- Dialogfenster
- Layout-Managers
- Events / Event-Handling
- **Parallel dazu Fachgespräche zur eigenen Arbeit (pro Person max 10')**

#### Nachbereitung

- GUI-Applikation (z.B. Wetter App, Quadratische Funktion) entwickeln

8) Daten aufbereiten und auswerten (Numpy, Matplotlib, Pandas, Excel)	
Datum	Di 17.01.2023
Zeit	17:30 – 20:45
Dozent	Walter Rothlin

#### Vorbereitung

- Numpy and Matplotlib API Documentation

#### Inhalte

- Objectbased programming mit Python: Access auf CSV und EXCEL Files
- Image-Processing mit Numpy und Matplotlib
- **Parallel dazu Fachgespräche zur eigenen Arbeit (pro Person max 10')**

#### Nachbereitung

- Applikationen fertig implementieren und testen.

9) AI in Python nutzen: OCR (optical character recognition)	
Datum	Di 31.01.2023
Zeit	17:30 – 20:45
Dozent	Walter Rothlin

#### Vorbereitung

- CV2 Library installieren.
- API und Examples studieren

#### Inhalte

- Eigene Anwendung implementieren
- **Fachgespräche zur eigenen Arbeit (pro Person max 10')**